Installation instructions for **lrgpr**

Dependencies and installation

Linux operating systems are supported, but unfortunately the optimizations used by **lrgpr** are not (easily) available on Windows or Mac OS X.

lrgpr has some dependencies that may not be installed on your system. Installation requires the GNU Scientific Library (GSL) and Boost C++ libraries.

- **Ubuntu / Debian:** these can be automatically installed with `apt-get`:

```
shell> sudo apt-get install libboost-all-dev libgsl0-dev
```

- **Redhat / CentOS:** these can be automatically installed with `yum`:

```
shell> sudo yum install boost-devel gsl-devel
```

Alternatively, you can install them manually from source:

- GSL: http://www.gnu.org/software/gsl/

- Boost: http://www.boost.org/users/download/

lrgpr also depends on a number of R packages that can be installed from CRAN:

```
> pkgs = c("Rcpp", "RcppGSL", "RcppProgress", "MASS", "doParallel",
"formula.tools", "BH", "bigmemory", "biganalytics", "aod")
> install.packages(pkgs)
```

lrgpr requires bigmemory >= v4.4.7, so install this from R-Forge:

```
> install.packages("bigmemory", repos="http://R-Forge.R-project.org")
```

Once the dependencies are installed, you can install **lrgpr**:

```
shell> R CMD INSTALL lrgpr_0.0.8.tar.gz
```

For very advanced R users

Accelerating linear algebra in R

Linear algebra operations are the bottleneck for **lrgpr** and many other packages in R. Compiling R with a good implementation of the BLAS linear algebra library can speed up some operations by >10x, and using a parallel library can further improve performance compared to the standard "reference BLAS" used by default. Using the best library for your machine, such as ATLAS (generic), ACML (AMD) or MKL (Intel), is strongly suggested.

You can follow the instructions to compile R against one of these libraries: http://cran.r-project.org/doc/manuals/R-admin.html#Linear-algebra